

9



web

programming

team

project

10

11

12

13

14

Trace_of_Time

15

```
<p style="text-align:left;">Left</p>
<p style="text-align:center;">Center</p>
<p style="text-align:right;">Right</p>
```

16

17

18

19

20

21

22

23

The_Trace_of_Time_Restored_by_AI
신경환 온인권 전지훈

24

25

26



AI가 복원하는 시간의 흔적

감정을 시각화하는 감성 AI 예술 프로젝트



web

programming

team

project

CONTENTS



01 프로젝트 개요

1-1. 개발동기

1-2. 기획의도

02 진행 프로세스

2-1. 팀원 및 주요역할

2-2. 기술스택

03 프로젝트 상세

3-1. 구조도

3-2. 기술설명

3-3. 주요코드 설명

04 프로젝트 시연

4-1. 기능화면

4-2. 시연영상

4-3. 마치며

9



web

programming

team

project

10

개발동기

11



- 인간의 삶은 태어난 이후부터 기록되기 시작
- 태어나기 이전의 시간, 역사 속 감정과 온도는 체감 어려움
- "AI가 사라진 시간의 흔적을 예술적으로 복원할 수 있을까?"
- Trace of Time 프로젝트 : 인간이 느끼지 못한 시간을 AI로 재해석 및 시각화

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26



web

programming

team

project

기획의도



- 초기 목표 : Stable Diffusion WebUI 활용 → 고정확도 이미지 복원
- 문제점 : 설치/환경 구성 복잡, 하드웨어 요구↑, 확장성·접근성 ↓
- 해결책 : OpenAI GPT + DALL-E-3 기반 이중 모델 아키텍처 전환
- 효과 : 텍스트 → 감정·상징 분석 → 시각적 이미지 재구성, 개발 효율성 및 정확성 확보



web

programming

team

project

팀원 및 주요역할

Full-Stack

신경환

- └ 시스템 아키텍처 설계
- └ DB 모델링 & 스키마 구축
- └ REST API 설계·개발
- └ AI 파이프라인 구축
- └ 텍스트 분석 모듈 개발
- └ 이미지 생성 모듈 개발
- └ 데이터 처리·저장 로직 개발
- └ 프론트엔드 UI/UX 개발

Back-End

온인권

- └ 코드 리뷰
- └ 버그바운딩
- └ JSON 데이터 수정
- └ 화면출력 수정

AI Prompt Engineering

전지훈

- └ AI 프롬프트 엔지니어링
- └ 세부 코드 디버깅
- └ 개발 서포트 엔지니어



web

programming

team

project

기술스택



Pyhton

프로그래밍 언어



FastAPI

프레임워크



ChatGPT

사용자가 입력한
텍스트 분석 지원



DALL-E

GPT 분석 결과 기반
이미지 생성



MySQL

데이터베이스



web

programming

team

project

구조도





web

programming

team

project

기술설명

Backend Architecture

FastAPI 라우터 설계

- ↳ /memory : 텍스트 기억 분석 + DB 저장
- ↳ /image : AI 이미지 생성 처리

MySQL DB 연동

- ↳ 저장 필드 : GPT 분석 결과, 이미지 결과 등 저장

AI 서비스 계층

- ↳ gpt_service.py : 텍스트 감정/상징/시대 분석
- ↳ dalle_service.py : AI 이미지 생성

이점

- ↳ 비동기 기반 FastAPI → 빠른 응답 속도
- ↳ 모듈형 구조 → 기능 확장성 용이
- ↳ API 기반 → 서비스형 플랫폼으로 발전 가능



web

programming

team

project

기술설명

Frontend Architecture

JavaScript + Bootstrap 기반 UI

index.html : 기억 입력 + AI 분석 요청

timeline.html : 기억 카드 리스트, 모달 상세보기

timeline.js

↳ 카드 동적 렌더링

↳ 모달 상세 표시

↳ 이미지 로딩 에러 핸들링

index.js

↳ Memory 저장 API 호출

↳ GPT 기반 기억 분석 요청, 이미지 생성 API 호출

↳ API 응답 기반 UI/UX 동작 처리

UX 설계 포인트

↳ 기억 + 감정 + 이미지를 한 카드로 묶어 저장

↳ 클릭하면 전체 스토리를 감상할 수 있도록 디자인

↳ 재복원 기능 → 같은 기억도 새로운 방식으로

다시 생성 가능



web

programming

team

project

기술설명

AI Model Architecture

GPT-5 : 텍스트 감정 분석

- └ 기억 속 감정 온도(긍정/부정/복합 감정)
- └ 상징성 요소 추출
- └ 스토리 맥락 분석 → 이미지 생성용 프롬프트 제작
- └ JSON 구조로 분석값 제공 → DB 저장 최적화

DALL-E-3 : 기억 복원 이미지 생성

- └ GPT 분석 결과 기반으로 이미지 생성
- └ 예술적 스타일로 기억을 복원한 이미지

GPT + DALL-E 연계 구조

- └ 백엔드가 두 모델을 순차 호출
- └ 텍스트 분석 → 이미지 생성 → DB 저장
- └ 분석 결과 + 이미지 URL 동시에 반환

9



web

programming

team

project

10

주요코드 설명(1)

11

gpt_service.py

12

- 사용자가 입력한 기억 텍스트를

13

감정적, 시각적, 상징적 정보로 분석

14

모델 : GPT-5

15

감정, 이미지, 시대, 상징

16

DB 저장과 이미지 생성 데이터 제공

17

18

19

20

21

22

23

24

25

26

```

try:
    response = client.chat.completions.create(
        model="gpt-5",
        messages=[
            {"role": "system", "content": system_prompt},
            {"role": "user", "content": text}
        ],
        temperature=1
    )
    result_text = response.choices[0].message.content

try:
    result_json = json.loads(result_text)

    analysis_summary = (
        f"감정: {result_json.get('emotion', '-')}\n"
        f"이미지: {result_json.get('imagery', '-')}\n"
        f"시대: {result_json.get('time_period', '-')}\n"
        f"상징: {result_json.get('symbolism', '-')}"
    )

    result_json["analysis"] = analysis_summary

except json.JSONDecodeError:

    result_json = {"analysis": result_text, "analysis_text": result_text}

return result_json

except Exception as e:
    logging.error(f"GPT 분석 실패: {e}")
    return {"analysis_error": str(e), "analysis": "분석 중 오류가 발생했습니다."}

```

9



web

programming

team

project

10

주요코드 설명(2)

11

dalle_service.py

12

- GPT 분석 결과를 기반으로

13

DALL-E-3 모델을 이용하여 이미지 생성

14

15

모델 : DALL-E-3

16

17

18

19

20

21

22

분석된 텍스트 데이터 → 프롬프트 변환

23

24

이미지 생성 후 URL 반환

25

26

```

def generate_image(prompt_or_gpt_result) -> str: 4개의 사용 위치

11     if isinstance(prompt_or_gpt_result, dict):
12         gpt_result = prompt_or_gpt_result
13         prompt = (
14             f"사용자의 기억을 사실적이고 감성적인 이미지로 재구성하세요.\n\n"
15             f"감정: {gpt_result.get('emotion', '')}\n"
16             f"감정 강도: {gpt_result.get('emotion_intensity', '')}\n"
17             f"장면 묘사: {gpt_result.get('imagery', '')}\n"
18             f"시각적 요소: {gpt_result.get('visual_elements', '')}\n"
19             f"색조/조명: {gpt_result.get('color_tone', '')}\n"
20             f"시간대/시대: {gpt_result.get('time_period', '')}\n"
21             f"장소/환경: {gpt_result.get('location_context', '')}\n"
22             f"상징적 의미: {gpt_result.get('symbolism', '')}\n"
23             f"중점 피사체: {gpt_result.get('visual_focus', '')}\n"
24             f"카메라 시점: {gpt_result.get('camera_perspective', '')}\n\n"
25             f"사진처럼 현실적이고 세부적인 묘사, 감정과 시간 흐름이 느껴지도록 구성하세요."
26         )
27     else:
28         prompt = str(prompt_or_gpt_result)

try:
    response = client.images.generate(
        model="dall-e-3",
        prompt=prompt,
        size="1024x1024"
    )

    image_url = response.data[0].url
    logging.info(f"이미지 생성 성공: {image_url}")
    return image_url

except Exception as e:
    logging.error(msg=f"이미지 생성 실패: {e}", exc_info=True)
    return ""

```



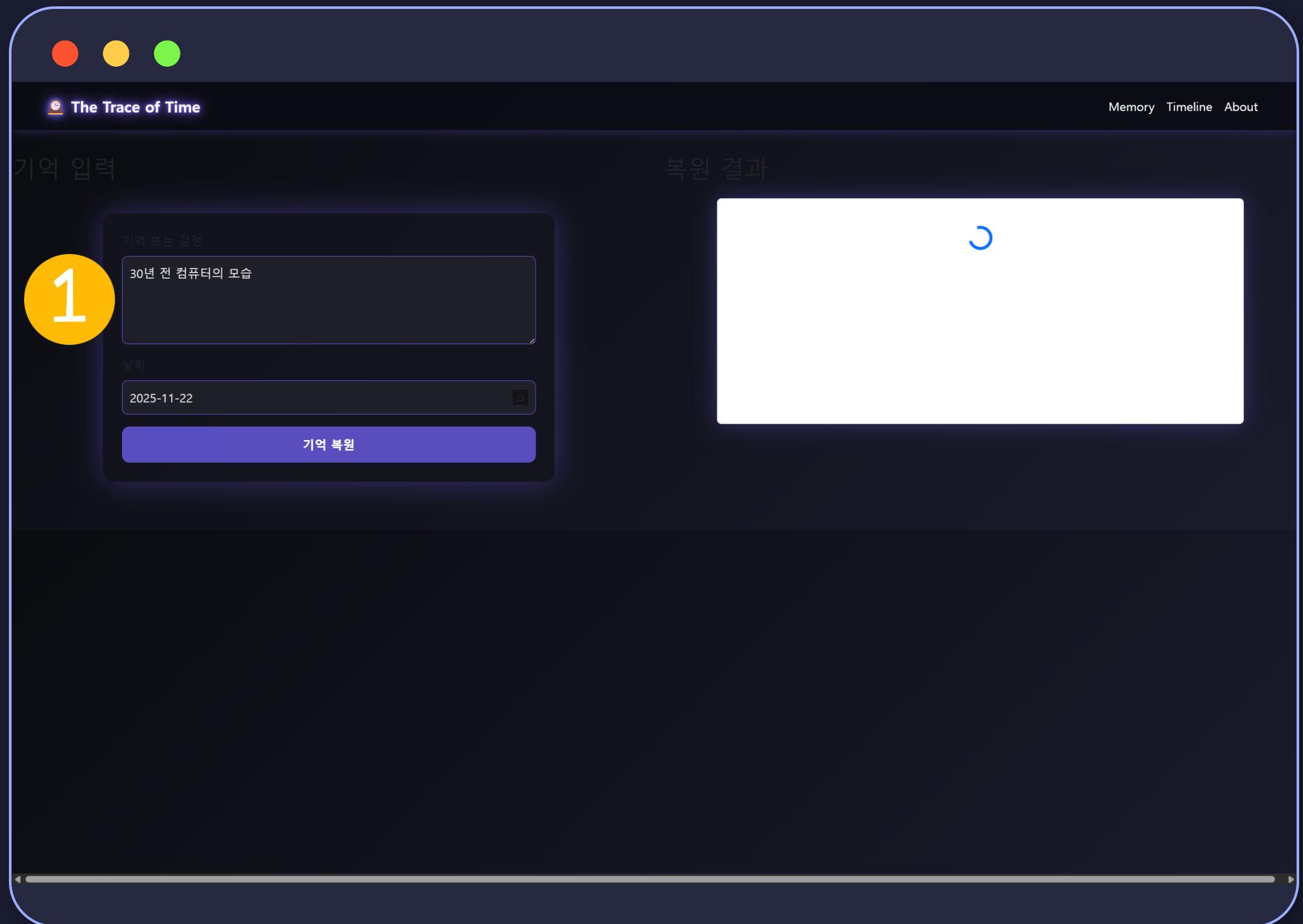
web

programming

team

project

기능화면



1

사용자가 기억을 입력



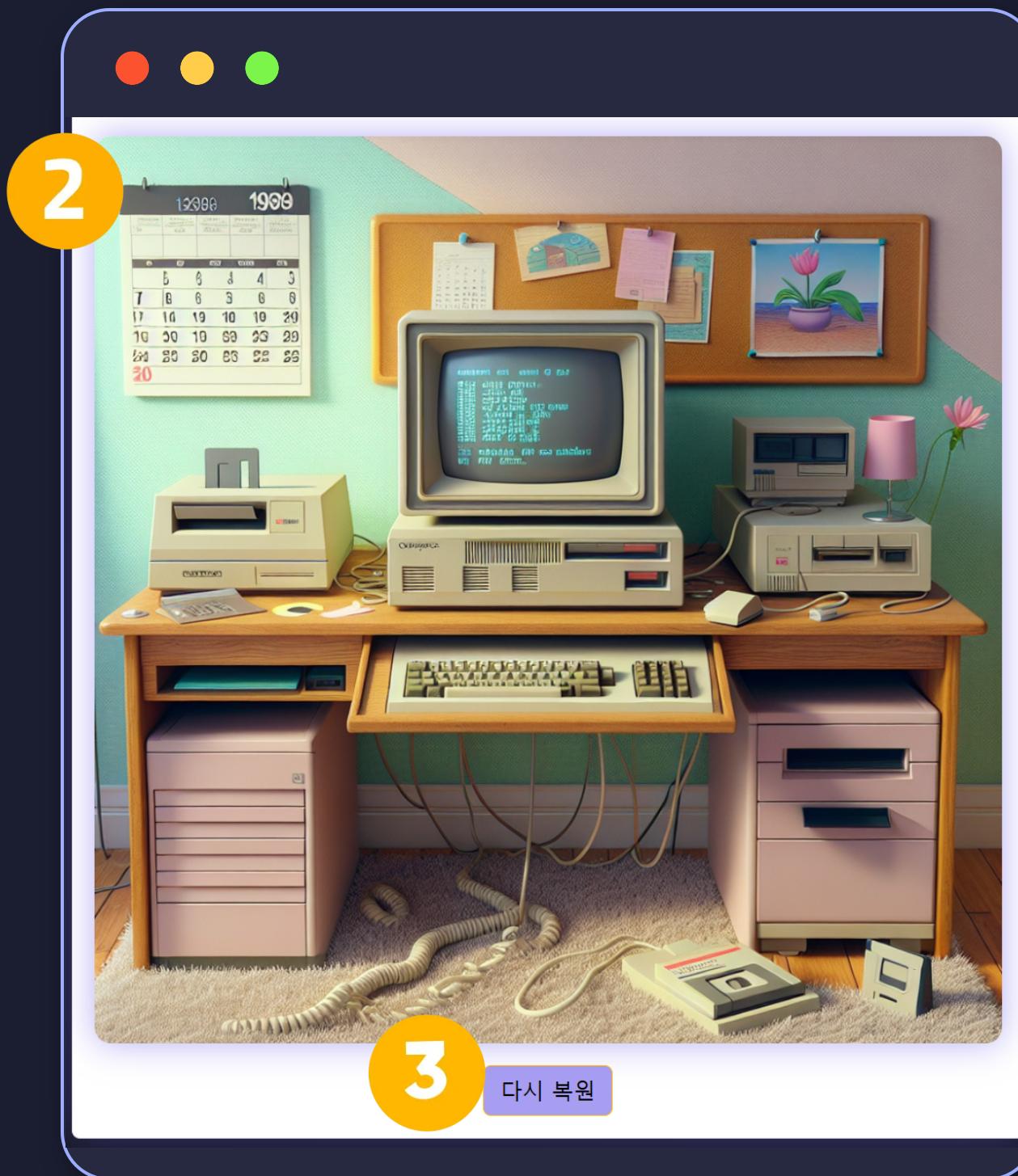
web

programming

team

project

기능화면



2

복원된 이미지 결과 출력

3

다시 복원 기능

9



web

programming

team

project

10

기능화면

11

4

30년 전 컴퓨터의 모습

감정: 그리움

이미지: 1990년대 중반, 서울의 작은 아파트 공부방. 벽지는 잔꽃무늬로 약간 누렇게 바랬고, 나무 무늬 합판 책상 위에 아이보리색 브라운관 모니터와 베이지 미들타워 본체가 놓여 있다. 모니터는 부팅 화면의 푸른빛을 내며 미세하게 흔들리고, 앞쪽 키보드의 키캡에는 살짝 번들거리는 사용 흔적이 물어 있다. 검은 고무볼 마우스가 파란 천 마우스패드 위에 놓여 있고, 옆에는 라벨이 붙은 3.5인치 플로피 디스크 몇 장이 가지런히 쌓여 있다. 책상 구석에는 모뎀의 초록·빨강 LED가 간헐적으로 깜빡이고, 코일선 유선 전화기가 모니터 뒤로 비스듬히 기대어 있다. 도트 프린터 옆으로는 연속지의 편침 훌 가장자리가 말려 있고, 사용 설명서와 두꺼운 소프트웨어 매뉴얼이 책받침처럼 쌓여 있다. 노란빛 스탠드가 책상을 비추고, 창문 너머엔 초겨울 저녁빛이 남아 있어 유리엔 얇은 성에가 끼어 있다. 창틀 위 머그컵에선 옅은 김이 오르고, 벽시계는 7시 42분을 가리킨다. 방 안엔 난방이 막 올라 따뜻하지만 공기 속에 먼지가 아주 약하게 떠, 모니터 빛에 반짝인다.

상징: 느림과 기다림(모뎀 LED의 깜빡임, 부팅 화면의 진행 표시), 연결의 시작(코일선 전화기와 모뎀), 손끝의 기억(번들거리는 키캡과 볼마우스), 시간의 누적(황변된 플라스틱과 쌓인 매뉴얼, 말려 있는 연속지)

시대: 1990년대 중반 초겨울 저녁

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

4

GPT가 분석한 결과 출력



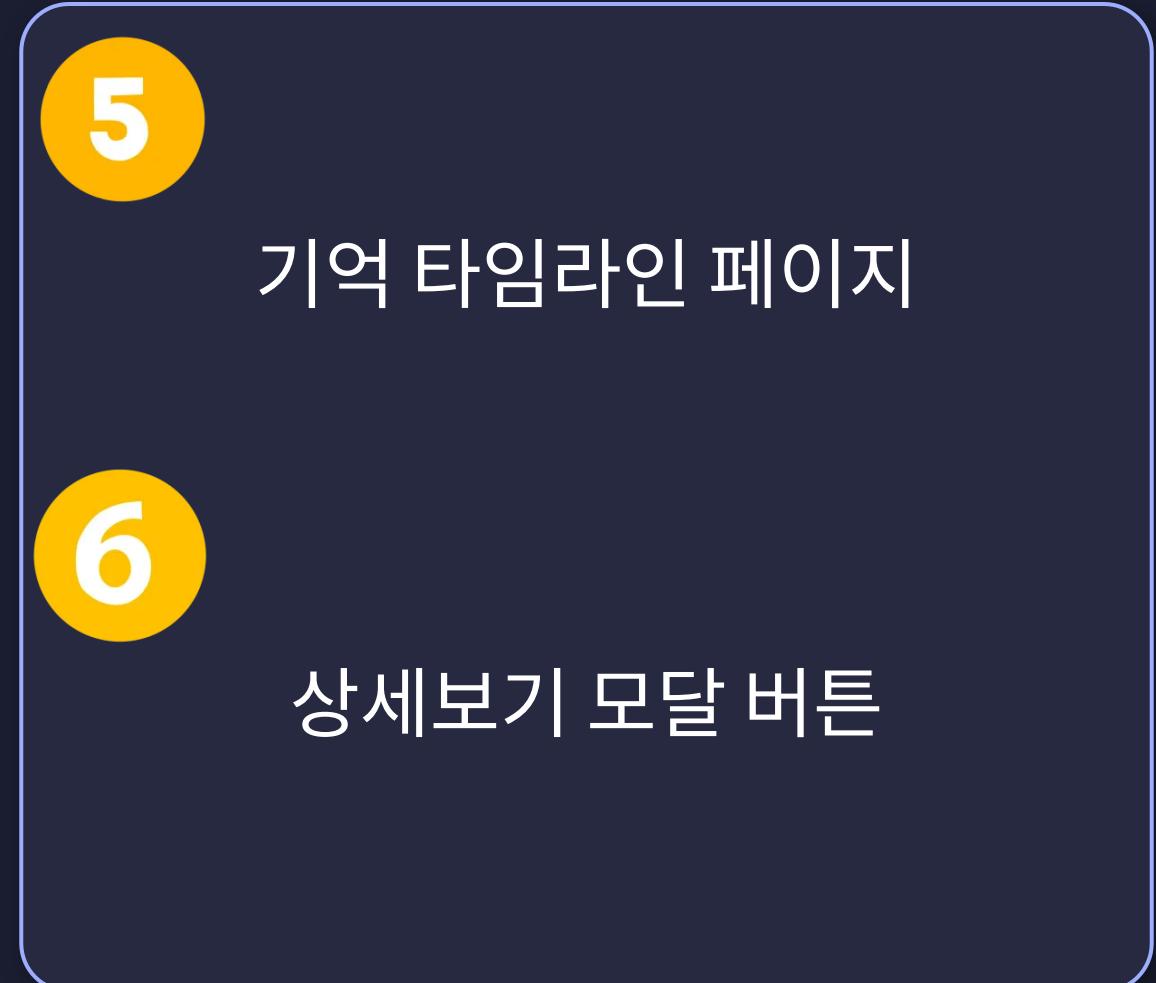
web

programming

team

project

기능화면





web

programming

team

project

기능화면



상세보기 모달

복원된 기억 수정 기능

복원된 기억 삭제 기능



web

programming

team

project

시연영상



<https://youtu.be/NguvJCP7qAc>

9



web

programming

team

project

10

마치며(결과)

11



사용자의 기억을 AI가 분석하고 이를 이미지로 시각화하여

12

새로운 형태의 개인 감정 기록 아카이브를 구축한 프로젝트

13

- 개인의 기억을 시각적으로 표현하는 새로운 경험 제공

14

- AI 멀티모달 파이프라인 구현(텍스트 → 감정 분석 → 시각화 이미지 생성)

15

- 직관적인 타임라인 기반 UI 제공

16

- 기억 수정 시 자동 재분석·이미지 재생성 기능 구현

17

- 개인 맞춤형 디지털 아카이브 생성

18

19

20

21

22

23

24

25

26



web

programming

team

project

9

10

마치며(느낀점)

11

12

신경환

13

14

- ↳ 이미지 생성 속도가 느리고 비용이 발생하는 문제 경험 → 최적화 및 성능 개선 필요성을 느낀
- ↳ 기능 추가 및 확장과 AWS 서버 배포의 미완성 아쉬움
- ↳ 웹 서비스의 모든 영역을 설계하고 개발하여 기술적으로도 조직적으로도 크게 성장

15

온인권

16

- ↳ Docker/K8s를 활용한 배포 환경 구축과 UI, AI 모델(Gemini) 선정 등 기술적 완성도 아쉬움
- ↳ 로그인 기반 개인화, 심층 조언 기능 미구현의 아쉬움

17

전지훈

18

- ↳ 추가매체(영상, 음악) 미구현 → 시각적 기능의 극대화 부족 → 개선의 필요성 느낀
- ↳ 이번 기회를 통해 설계 영역에서 크게 성장

19

20

21



web

programming

team

project

마치며(Q&A)



자유롭게 질문해주세요!



web

programming

team

project



감사합니다!

The_Trace_of_Time_Restored_by_AI

신경환 | 온인권 | 전지훈